Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 93 (2013) 1461 – 1466

Procedia
Social and Behavioral Sciences

3rd World Conference on Learning, Teaching and Educational Leadership – WCLTA 2012

A constructivist approach for implementing and evaluating algorithms in a machine vision course at the undergraduate level

Iván García [1], Enrique Guzmán-Ramírez, Carlos Gonzalez-Rojas.

Universidad Tecnologica de la Mixteca, Carretera a Acatlima Km. 2.5, Oaxaca 69000, Mexico

Abstract

The growth of areas such as automation and robotics demands autonomous systems endowed with sophisticated perception systems like machine vision. However, undergraduate level education is not providing good results in this sense, because students do not participate in the creation of their own knowledge; they are only passive observers. On the other hand, FPGA technology has great potential in areas such as machine vision where many hypotheses are evaluated concurrently, high computation rates are required, and the current systems are far from equaling human performance. Our research notes that by using FPGA and constructivist learning as the methodology, assessment of learning in electronic sciences is not a separate process after learning has occurred, but rather learning and assessment are coterminous. Thus, this paper presents a constructivist tool focused on a machine vision course that allows students to implement and evaluate algorithms of this area on an FPGA.

Selection and peer review under responsibility of Prof. Dr. Ferhan Odabaşı

Keywords: Constructivist; video processing; undergraduate level; FPGA, process improvement.

1. Introduction

Undergraduate education of professional engineers in areas of computer sciences and electronics requires effective methods for teaching and learning. According to Azis (2011), Tsai et al. (2011), and Shyr (2012), undergraduate engineering education has changed over the last years because the integration of Information Technology (IT) into the classrooms has provided innovative teaching/learning environments to students. In this sense, Weng et al. (2009) state that student retention has been an imperative research topic in undergraduate education since the 1970's.

However, although the issue of student retention has been investigated for decades, more research has specifically examined IT interaction among students and its use for their own knowledge creation. Thus, the constructivist paradigm has been widely explored as support in the computer science area (Alonso et al., 2009; Appavoo, 2010; Huang et al., 2010), because learners take an active role in their learning, since they not only retain information, but also connect it with previously assimilated knowledge to construct their new knowledge. Various researchers have been interested in applying the constructivist approach to the artificial intelligence area (i.e., Hurst & Bull, 2006; Samper et al., 2010).

In this context, the machine vision field attempts to understand the frame video information, generally formed by high-dimensional data from the real world, in order to produce decisions in the form of simple numerical or

1Corresponding Author: Ivan Garcia. Tel.: +52-953-5320214
E-mail address: ivan@mixteco.utm.mx

Selection and peer review under responsibility of Prof. Dr. Ferhan Odabaşı
doi:10.1016/j.sbspro.2013.10.064

symbolic information. As a technological discipline, machine vision seeks to apply its theories and models to the construction of systems that meet these requirements. Examples of applications of machine vision include systems for stereo vision, detecting events for visual surveillance, navigation of an autonomous vehicle or mobile robot, modeling of objects or environments, automatic inspection, interaction with environments, etc.

Many of these applications need to be implemented in autonomous systems. In implementing such systems, several approaches can be distinguished, such as the DSP-based (Illgner, 2000), DSP-FPGA-based (Christophersen et al., 2004; Fresse et al., 2002), and FPGA-based (Price et al., 2006).

Machine vision systems include stages for acquiring, processing, analyzing, and understanding images. In each of these stages, a great amount of theories and models have been proposed and documented. The implementation of new models or variants of existing ones on autonomous systems for undergraduate education requires specialized tools that enable the student to analyze his/her behavior and evaluate his/her performance. Although many tools have been designed using a DSP as central processing unit (Guan et al., 2007; Lv et al., 2009), using a hybrid approach DSP-FPGA (Jianhua et al., 2007; Jinghong et al., 2007; Yan et al., 2006) or using an FPGA-based approach (Toledo et al., 2007; Haynes et al., 2002; Cuadrado et al., 2006), there is no research related to incorporating the constructivist approach to support the learning process. Moreover, the existing tools show a deficiency; none of them provide a method for introducing the proposed tool in the context of an educational program for higher education. This paper presents a tool focused on a machine vision course which allows students to implement and evaluate algorithms of this area on a FPGA. The tool includes an algorithms repertory which can be tailored to the topic under study. Furthermore, the tool enables that the algorithms repertory can be easily expanded and existing algorithms modified through the constructivist approach.

2. Constructivist approach for a machine vision course

During the last 10 years, different universities have adopted IT to implement alternative methods for learning and teaching. Considering this influence and recognizing the impact of technological development on society, universities around the world are currently using a new set of technologies for communicating and computing, and they are discovering their potential to improve strategic effectiveness in teaching. Amongst them, one line of research that has produced positive results makes use of the *constructivist* method of teaching. The constructivist theory was initially conceived by Jean Piaget as a result of his research in how children construct their knowledge. Piaget developed many theories, describing the stages of a child's cognitive development, and, supported by his extensive research work (Piaget, 1970), he established an analysis methodology that set the basis for his learning theory: the genetic epistemology. In this context, various researchers have focused on various aspects of this approach (leading to variants like personal and social constructivism, or radical and pragmatic constructivism), but one of the most important authors is Von Glasersfeld. He discussed radical constructivism as a theory of knowledge and cognition and its applications for teaching (Von Glasersfeld, 2006). In pedagogic models based on the constructivist theory, students should construct their own knowledge instead of passively absorbing it in a classroom or by consulting textbooks. Our idea is to develop interactive mechanisms within a software/hardware tool under the constructivist model (see Figure 1).

This way of learning demands that the undergraduate student not only discovers the facts, but also creates mental models from them that result in the knowledge construction. The tasks for monitoring and stimulating the students to achieve their objectives are assigned to teachers, who should be, at the same time, conscious of the individual cognitive structures of each student, which makes the method pedagogically more complex than the classical method (principally based on expository lessons without interaction). The constructivist model recognizes the benefit achieved when students participate in video processing tasks that enable the active construction of their own knowledge domain. In order to do this, we propose the development of a teaching tool based on solid grounding in Piagetian fundamentals. This software/hardware tool attempts to improve student interaction in a machine vision course, in contrast to conventional teaching techniques in higher education.

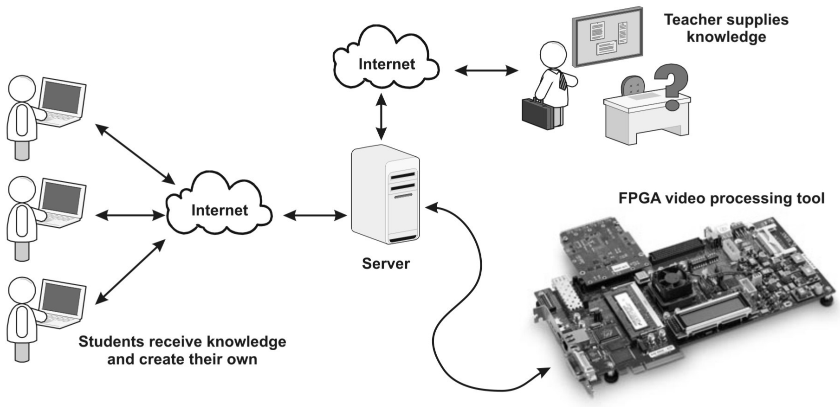

Figure 1. FPGA-based tool for video processing

3. The FPGA-based tool for video processing

This tool has been focused on educational and research fields associated to machine vision and includes an efficient and fast design process of video processing algorithms on FPGAs, enabling the modeling, implementation and performance evaluation of the algorithms, propitiating in this way the active construction of knowledge. The proposed tool consists of a *FPGA-based Video Capture and Processing System (Fb-VCPS)* and *Application Interface* (see Figure 2), each one designed to accomplish a specific function.

3.1. FPGA-based Video Capture and Processing System

The Fb-VCPS is an embedded system based on Xilinx Spartan-6 LX16 FPGA. The SPIES methodology proposed by Garcia and Herrera (2010) was used for the design and implementation of the *Fb-VCPS*. The central component of Fb-VCPS, named Manager of Hardware Applications, is responsible for supervising the execution of video processing algorithms and management of the system resources. The components that complement this system are: a Video Capture System, a USB Controller for interfacing to Application Interface, and the Memory System (see Figure 2). The USB Controller consists of the DLP-USB1232H device; it is a USB to parallel FIFO interface module based on FT2232HQ a FTDI's 5th generation. The Memory System is used as an external temporary buffer to store video frames to be processed and the processing results; this system is composed of a Micron M45W8MW16 Cellular RAM device organized as 128 Mb: 8 Mb × 16 bits. The Video Capture System features an Aptina MT9D112 2-megapixel CMOS digital image sensor and an image flow processor (IFP).

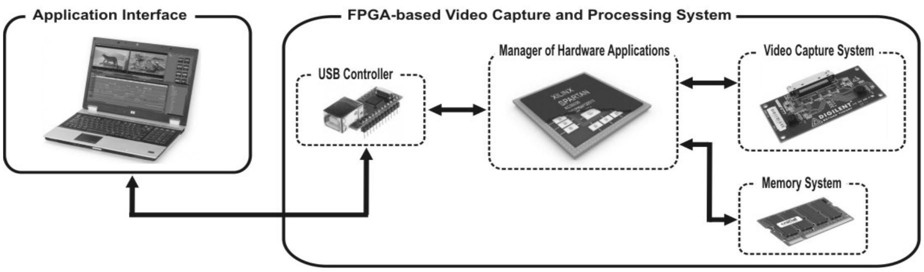

Figure 2. FPGA-based tool for video processing.

3.1.1. Manager of hardware applications

The Manager of Hardware Applications is an element embedded on FPGA; its description is founded on a hierarchical and modular approach using top-down methodology (Palnitkar, 2003; Pardo & Boluda, 2004). For this design, the modules were described following the dataflow and behavioral levels modeling using the VHDL hardware description language standard (IEEE Computer Society, 1988) and the Electronic Design Automation (EDA) tool Xilinx ISE Design Suite 12.2.

By applying the methodology referred, an initial modular partitioning step generates seven modules: Clock Manager, USB Driver, Memory System Driver, Control Unit, Register File, Repository of Algorithms and Video Capture System Driver, all of them configured into FPGA. The principal element of the Clock Manager module is a Digital Clock Manager (DCM). DCM is an embedded element on the Spartan-6 FPGA, DCM allows to multiply or divide the incoming clock frequency to synthesize a new clock frequency and also eliminates clock skew, thereby improving system performance. From the principal 100MHz clock, the Clock Manager module generates the global clock system (`Clk`) for the Manager of Hardware Applications providing a correction clock feature and ensuring a clean `Clk` output clock with a 50% duty cycle.

The USB Driver module, based on the timing diagram of the DLP-USB1232H device, describes the interface protocol in a finite-state machine that allows the Manager of Hardware Applications to access the resources of USB controller and to establish the fast transferring data with the Application Interface.

The Memory System Driver module is described as a finite-state machine that implements the access protocol with the M45W8MW16 Cellular RAM. This module provides an efficient interface for the *Memory System* when a module requests a write or read operation.

The Register File is a LUT-based module; this module includes ten 24-bit general purpose registers; it is used to store information to be shared by Control Unit, Video Capture System Driver and Algorithms modules.

The Control Unit module, implemented through a finite-state machine, governs the operation and interaction among all components that integrate the *Fb-VCPS*.

The Repository of Algorithms is a set of basic image and video processing and analysis algorithms that the proposed tool includes; any of these algorithms can be easily removed. Moreover, the tool enables that the algorithms repertory can be expanded or modify existing ones, enabling the students' own knowledge creation.

The Video Capture System Driver module allows the Manager of Hardware Applications to communicate with the different sub-systems of the Video Capture System. Due to this, the Video Capture System Driver module is divided of 3 sub- modules hierarchically inferior, they are: *I*2C Interface, Camera Control and Parallel Input Controller. The I2C Interface sub-module implements a bi-directional serial bus that provides a simple and efficient method of data exchange between Camera Control and Fb-VCPS. Furthermore, this sub-component is required for configuration and setting of imaging parameters, and enables read/write access to control and status registers within the MT9D112. The Camera Control sub-module is described as a finite-state machine that implements the sequences needed and configurations to trigger the Video Capture System. Finally, the Parallel Input Controller sub-module allows reading data frames from video capture system and generates signals to store them into the Memory System.

3.2. Application Interface

The goal of developing tools based on embedded systems is to integrate services that enable the user access to all the resources offered by the tool in a convenient and comfortable manner. The graphical user interface (GUI) of embedded software plays an important role in reaching this goal. The purpose of providing a GUI is to offer the user an efficient way to communicate with the tool. The Application Interface is a GUI; it is a software application running on the host PC. To support the evaluation and evolution, *Application Interface* was developed based on C++ following the method proposed by Yang et al. (2007), where they introduced a new mini visual IDE, called VY. The Application Interface allows the user to select the configuration that will be used during the capture of a video sequence, enable video capture, select which algorithm will be applied on the captured video frame and enable visualization of video processing results.

4. Experimental results

For this purpose, we present an example and results of its implementation by students using the tool. In this example, a video frame is captured by students and four algorithms are applied simultaneously: *negative*, *thresholding*, *edge detection* and *grayscale covert* (see Figure 3). The first example is the negative. The negative is defined as $s = (L-1) - r$; where L is the number of bits used to represent each component, $r = \{R, G, B\}$ and represents the value of each component, $s = \{R, G, B\}$ and represents the negative of each component.

Thresholding is the second example and it is the simplest method of image segmentation. Thresholding can be used to create binary images. The idea for this work is simple, one must decide which color tone to give each pixel when it is greater than a threshold (threshold value), and the remaining pixels have by default the other color tone.

The third example is edge detection. Edge detection is the most common technique for detecting significant discontinuities in the intensity values of an image. These discontinuities are detected using the first and second derivatives. The choice of the first order derivative is processing the image gradient. The gradient of a 2-D function $f(x, y)$ is defined as the following vector:

$$\nabla f = \begin{pmatrix} Gx \\ Gy \end{pmatrix} = \begin{pmatrix} \partial f / \partial x \\ \partial f / \partial y \end{pmatrix}$$

In order to reduce the complexity of the implementation, an approximation of gradient magnitude is used to implement the edge detection algorithms, $\nabla f \approx |Gx| + |Gy|$. The fundamental property of the gradient vector is always pointing in the direction of the maximum index of change of the function $f(x, y)$ at coordinates (x, y).

The last example is the RGB color to grayscale conversion. When each color pixel is described by a triple (R, G, B) of intensities for red, green, and blue, several methods for converting color to grayscale can be used. We use the lightness method; this method averages the most prominent and least prominent colors: $(\mathbf{max}(R, G, B) + \mathbf{min}(R, G, B))/2$.

5. Conclusions

The constructivist theory provides effective theoretical guidance to the interactive course design and the development of undergraduate teaching sources for computer science areas. In this paper we investigate how the adoption of a constructivist approach opens excellent perspectives for improvements in universities' teaching-learning process, specifically related to high technical courses that require a lot of interaction between teacher and students, like the machine vision course. The FPGA-based tool described above provides an illustrated and simplified mechanism to process and analyze algorithms for video processing. This tool encourages undergraduate students to construct knowledge actively and intentionally in authentic scenarios.

References

Alonso, F., Manrique, D., & Viñes, J. M. (2009). A moderate constructivist e-learning instructional model evaluated on computer specialists. *Computers & Education*, *53*(1), 57-65.

Appavoo, P. (2010). The constructivist' approach to teaching computing. *International Journal of Continuing Engineering Education and Life Long Learning*, *20*(3-5), 407-417.

Aziz, E-S. (2011). Teaching and learning enhancement in undergraduate machine dynamics. *Computer Applications in Engineering Education*, *19*(2), 244-255.

Christophersen, H. B., Pickell, W. J., Koller, A. A., Kannan, S. K., & Johnson, E. N. (2004). Small adaptive flight control systems for UAVs using FPGA/DSP technology. *Proceedings of the AIAA "Unmanned Unlimited" Technical Conference, Workshop and Exhibit,* 245-270. doi: 10.2514/MUAV04

Cuadrado, C., Zuloaga, A., Martin, J. L., Lazaro, J. & Jimenez, J. (2006). Real-time stereo vision processing system in a FPGA. *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics*, 3455-3460. doi: 10.1109/IECON.2006.347755

Design Automation Standards Subcommittee of the Design automation technical committee of the IEEE computer society (1988). *IEEE standard VHDL language reference manual*, IEEE Std 1076-1987, Institute of Electrical and Electronics Engineers.

Jinghong, D., Yaling, D., & Kun, L. (2007). Development of image processing system based on DSP and FPGA. *Proceedings of the 8th International Conference on Electronic Measurement and Instruments 2*, 791-794. doi: 10.1109/ICEMI.2007.4350799

Fresse, V., Déforges, O., & Nezan, J-F. (2002). AVSynDEx: A rapid prototyping process dedicated to the implementation of digital image processing applications on multi-DSP and FPGA architectures. *Journal on Applied Signal Processing*, *9*, 990-1002.

Garcia, I., & Herrera, A. (2010). Using the software process improvement approach for defining a methodology for embedded systems development using the CMMI-DEV v1.2. *Proceedings of the 10th International Conference on Computer and Information Technology,* 233-240. doi: 10.1109/CIT.2010.74

Guan, Y., Luo, L., Li, Z., & Wu, M. (2007). Research for the video image processing system experiment based on DSP. *Chinese Journal of Electron Devices*, *7*(2)*, 152-163.*

Haynes, S. D., Epsom, H. G., Cooper, R. J., & McAlpine, P. L. (2002).UltraSONIC: A reconfigurable architecture for video image processing. *Field-Programmable Logic and Applications, Lecture Notes in Computer Science*, *2438*, 25-45.

Huang, H-M., Rauch, U., & Liaw, S-S. (2010). Investigating learners' attitudes toward virtual reality learning environments: Based on a constructivist approach. *Computers & Education*, *55*(3), 1171-1182.

Hurts, J., & Bull, L. (2006). A neural learning classifier system with self-adaptive constructivism for mobile robot control. *Artifical Life*, *12*(3), 353-380.

Illgner, K. (2000). DSPs for image and video processing. *Signal Processing*, *80* (11), 2323-2336.

Yan, L., Zhang, T., & Zhong, S. (2006). A DSP/FPGA - Based parallel architecture for real-time image processing. *Proceedings of the 6th World Congress on Intelligent Control and Automation 2,* 10022-10025. doi: 10.1109/WCICA.2006.1713959

Lv, Q., Liu Z., & Wang K. (2009). Design of real-time image processing system based on TMS320DM642/Matlab. *Electronic Design Engineering*, *12*(4), 450-472.

Palnitkar, S. (2003). *Verilog HDL: A guide to digital design and synthesis*. New Jersey: Prentice-Hall.

Pardo, F., & Boluda J. A. (2004).*VHDL lenguaje para síntesis y modelado de circuitos*. (2nd ed.). España : RA-MA.

Piaget, J. (1970). *Genetic epistemology*. New York: Norton.

Price, A., Pyke, J., Ashiri, D., & Cornall, T. (2006). Real time object detection for an unmanned aerial vehicle using an FPGA based vision system. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation 2,* 2854-2859. doi: 10.1109/ROBOT.2006.1642134

Samper, D., Santolaria, J., Pastor, J. J., & Aguilar, J. J. (2010). Teaching camera calibration by a constructivist methodology. *IEEE Transactions on Education*, *53*(4), 646-652.

Shyr, W-J. (2012). Teaching mechatronics: An innovative group project-based approach. *Computer Applications in Engineering Education*, *20*(1), 93-102.

Toledo, F. J., Martinez, J. J., & Ferrandez, J. M. (2007). FPGA-based platform for image and video processing embedded systems. *Proceedings of the 3rd Southern Conference on Programmable Logic,* 171-176. doi: 10.1109/SPL.2007.371743

Tsai, W-T., Li, W., Elston, J., & Chen, Y. (2011). Collaborative learning using wiki web sites for computer science undergraduate education: A case study. *IEEE Transactions on Education*, *54*(1), 114-124.

Von Glasersfeld, E. (2006). A constructivist approach to experiential foundations of mathematical concepts revisited. *Constructivist Foundations*, *1*(2), 61-72.

Jianhua, W., Chanlao, L., Dachuan, C., & Yangguang, Z. (2007). Design of real-time video processing system based on DSP+FPGA. *Foreign Electronic Measurement Technology*, 09.

Weng, F., Cheong, F., & Cheong, C. (2009). The combined effect of self-efficacy and academic integration on higher education students studying IT majors in Taiwan. *Education and Information Technologies*, *15*(4), 333-353.

Yang, L., Choi, Y., Seo, C., Yang, T., and Kim, M. (2007). Design of VY: a mini visual IDE for the development of GUI in embedded devices. *Proceedings of the 5th ACIS International Conference on Software Engineering Research, Management and Applications*, 625–632. doi: 10.1109/SERA.2007.76